

INCO: International Scientific Cooperation Projects (1998-2002)

**SUSTAINABLE WATER MANAGEMENT IN
MEDITERRANEAN COASTAL AQUIFERS:
Recharge Assessment and Modelling Issues
(SWIMED)**

Contract number **ICA3-CT2002-10004**

Automatic calibration of CODESA-3D using PEST

Giuditta Lecca

CRS4 contribution to Deliverable 8 "Optimization & Socio-Economic Models"



December, 31 2005

Contents

Automatic calibration of CODESA-3D using PEST	3
Introduction	3
PEST	4
The groundwater model	4
The optimization model	5
Synthetic observations as control data	6
The CODESA-3D-PEST interface	7
Results	11
Concluding remarks and further applications	15
Bibliography	15

List of Figures

Figure 1. Work-flow of the simulation and calibration process for groundwater 3D model.	3
Figure 2. Sketch of the overall optimization model (PEST + CODESA-3D).	5
Figure 3. Batch file <i>doit</i> called by PEST as the embedded model.	6
Figure 4. CODESA-3D input file <i>soil</i> containing the selected adjustable parameters.	7
Figure 5. Example of intermediate input file (<i>Oristano-K.inp</i>).	7
Figure 6. Example of PEST template file (<i>Oristano-K.tpl</i>).	8
Figure 7. Example of CODESA-3D output file (<i>psi.out</i>).	8
Figure 8. Example of intermediate output file (<i>psi-control.out</i>).	8
Figure 9. Example of PEST instruction file (<i>Oristano-K.ins</i>).	9
Figure 10. Example of the PEST control file used for the Oristano case study.	10
Figure 11. Extract of the PEST run record file.	12
Figure 12. Some optimization results from the PEST run record file Oristano-K.rec.	13
Figure 13. Parameter covariance and correlation coefficient matrices.	14
Figure 14. Part of the parameter sensitivity file (<filename>.sen).	15
Figure 15. Part of the observation sensitivity file (<filename>.seo).	15

Automatic calibration of CODESA-3D using PEST

Giuditta Lecca

Hydrology and Water Resources Management
Centre for Advanced Studies, Research and Development in Sardinia (CRS4)

Introduction

We describe here our experience in using the *Model Independent Parameter ESTimation* (PEST) free software tool [Doherty, 2002] to perform the automatic calibration of the *COupled DEnsity-dependent variably SATurated flow and miscible transport* (CODESA-3D) groundwater model [Gambolati et al., 1999]. Generally speaking, calibration of a model requires that a suitable method of spatial parameter characterization be defined in order to adjust model parameters until model outputs correspond well to specific laboratory and/or field measurements of the system which is simulated. In particular, for groundwater models the adjustable parameters are usually given by main hydrogeological properties (e.g. hydraulic permeability) and/or system excitations (e.g. abstraction volumes) while control data are represented by piezometric heads and/or salt concentrations measured in the field. Model calibration is a complex task. To perform it for a 3D fully-distributed physically-based hydrological model we need to build up a chain of interdependent software tools and data through the interdisciplinary expertise of GIS experts, modelers and hydrogeologists (Figure 1).

The newly generated optimization model is comprised by the two pieces of software CODESA-3D and PEST with the latter wrapping the former up. The optimization model is not restricted in its use solely to the calibration of the groundwater model, through this tool modeler can gain valuable insight into the strengths and weakness of the input dataset allowing future data gathering to be undertaken in an optimal manner. In addition, lessons learned will be applicable also to the estimation of the degree of uncertainty associated with a given calibrated model prediction and to make decisions regarding appropriate levels of model complexity.

In the following we discuss in detail the optimization model development and test using “synthetic observations” generated by the groundwater model itself.

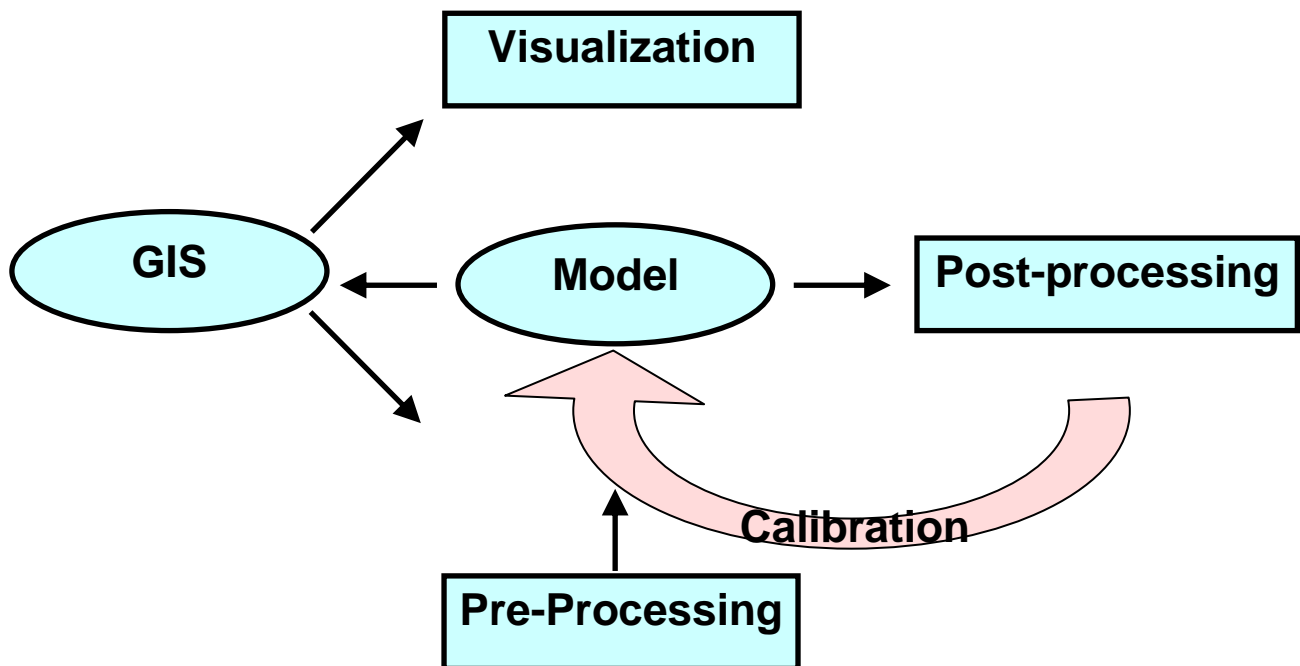


Figure 1. Work-flow of the simulation and calibration process for a fully-distributed physically-based groundwater 3D model.

PEST

PEST is a nonlinear model-independent parameter estimation package that can be used to estimate parameters for any existing model even if the user do not have the model source code. PEST is currently being used in many field of science and engineering and it has become a groundwater industry standard. Indeed some of the most popular computer codes like SWAT¹, MODFLOW², MT3D³, GMS⁴, etc, use it to implement automatic calibration modules.

PEST adjust model parameters in order to reduce to a minimum the discrepancy between model-generated outputs and the corresponding field measurements. The computer code does it taking the control of the embedded model and running it as many times as is necessary in order to determine the optimal set of model parameters in a weighted least square sense. PEST uses a nonlinear technique known as the Gauss-Marquardt-Levenberg method whose strength is to require fewer model runs than any other estimation method. A mathematical requirement of the GML approach is that the dependence of model-generated observations on adjustable model parameters be continuously differentiable.

Another software requirement of PEST is that model input and output variables involved in the optimization process are written in ASCII (i.e. text) files. In our case this was easily achieved using pre- and post-processing tools to link together PEST with CODESA-3D without modifying directly any original model input/output formats.

For linear models the GML algorithm can give the optimal parameter set just in one iteration while for nonlinear models (CODESA-3D falls under this category) parameter estimation is an iterative process. At each iteration step the relationship between selected model parameters (inputs) and model-generated observations (outputs) is linearized by means of the Taylor expansion about the actual best parameter set, hence the derivatives of all outputs with respect to all parameters are calculated. Then the linearized problem is solved for a better parameter set and this set is tested by a new model run. The iterative procedure is stopped when the objective function, generally speaking the sum of squared deviations between model outputs and field measures, reduces to a minimum corresponding to a user-defined threshold.

As the calibration process proceeds, PEST continuously records the sensitivity of each adjustable parameter to the observation dataset which is available for user inspection. Through this information the modeler can discover those parameters that influence mostly the calibration and those that are practically not relevant to it. In addition at the end of the process, PEST writes a large amount of useful auxiliary data (parameter correlation coefficient matrix, parameter covariance matrix etc.). All this information helps a lot the modeler to refine the conceptual model of the system under study, to exploit available data and to efficiently plan future campaign of data acquisition.

Besides the traditional way of using PEST in performing parameter estimation (model calibration), predictive analysis mode determines the maximum/minimum model predictions while still maintaining the model calibrated below a user-defined threshold. This mode allows the user to assess upper and lower bounds of the uncertainty interval associated with model predictions, which is definitely the true added value of any modeling analysis.

Another important tool of the PEST software distribution is the Parallel PEST (PPEST) module. It distributes model runs across networked PCs. Where model run times are large and adjustable parameters are many, the saving in overall optimization time through the use of the parallel module can be enormous. Thus Parallel PEST can be used in the calibration of large and complex models where application of nonlinear parameter estimation techniques would have previously been considered impossible.

The groundwater model

CODESA-3D is a fully-distributed physically-based hydrological model. In more details CODESA-3D is a three-dimensional finite element simulator for groundwater flow and solute transport in variably saturated porous media on unstructured domains. The flow and solute transport processes are coupled through the variable density of the filtrating mixture made of water and dissolved

¹ <http://www.brc.tamus.edu/swat/>

² http://www.scisoftware.com/products/modflow_details/modflow_details.html

³ <http://www.epa.gov/ada/csmos/models/mt3d.html>

⁴ <http://chl.erdc.usace.army.mil/CHL.aspx?p=s&a=Software;1>

matter (salt, pollutants). The flow module simulates the water movement in the porous medium, taking into account different forcing inputs: infiltration/evaporation, recharge/discharge, withdrawal/injection, etc., while the transport module computes the migration of the salty plume due to advection and diffusion processes. Model parameters and system excitations are assumed variable in space and/or time (for details see SWIMED project Deliverable D6 *Groundwater Flow Models* [Ababou and Al-Bitar, 2005] and D7 *Seawater Intrusion Models* [Lecca and Larabi, 2005]. Typical applications of the model are so-called density-dependent problems in subsurface hydrology; in particular the model has been applied to the saltwater intrusion problem of coastal aquifer. Denser-than-water non-aqueous phase liquids (DNAPLs), such as chlorinated organic contaminants, are other examples of density-dependent contaminants, which can be modeled with CODESA-3D. Recently the code has been coupled with a genetic algorithm to compute optimal pumping volumes for an hypothetical aquifer under constraints by Palestinian and Moroccan SWIMED partners [Qahman et al., 2005]. CODESA-3D computer code can be obtained from CRS4 (Italy) through a license agreement for research purposes only. The code manual is *Implementing and Testing CODESA-3D* [Lecca, 2000].

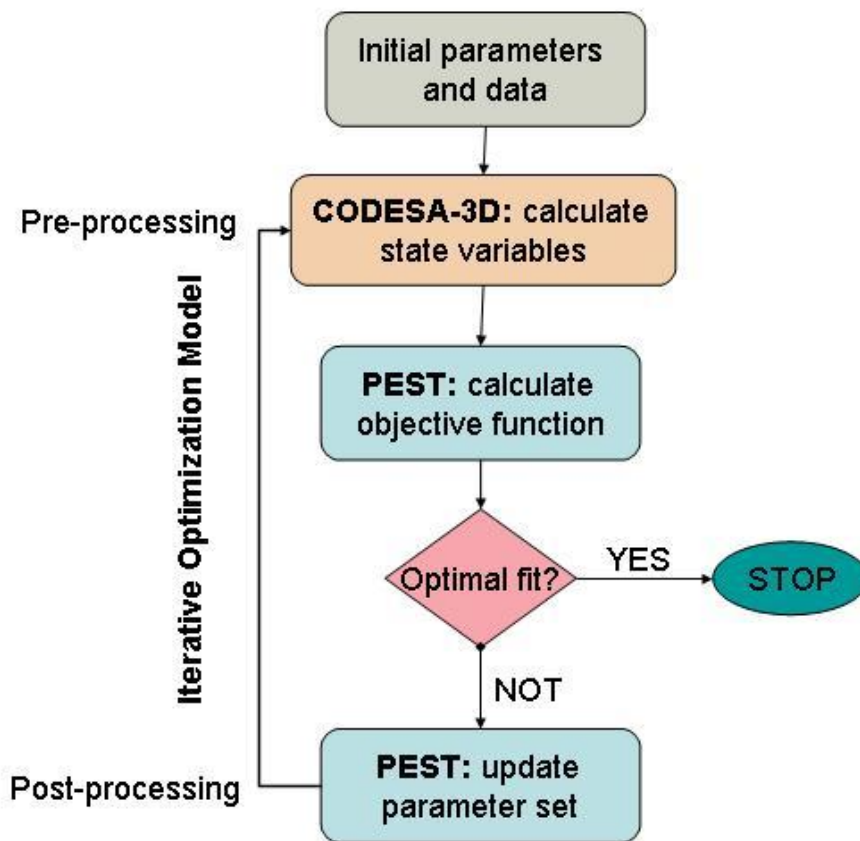


Figure 2. Sketch of the overall optimization model (PEST + CODESA-3D).

The optimization model

The optimization model is represented by the integrated system, shown in Figure 2, between executables (models and tools) and datasets performing an iterative calibration process. In particular, CODESA-3D outputs feed a post-processor that extracts pertinent data from large model files and put them into smaller text files for easier PEST access. Hence PEST computes the objective function and, if required, calculates the next optimal parameter set. PEST outputs are written in a small file which in turn is pre-processed to generate appropriate CODESA-3D input files. This process results in a single batch job (Figure 3) made of a chain of 3 executables in succession (“composite model”). The first processor (*write_soil*) reads a PEST output file (current

adjustable parameter set) and write the corresponding CODESA-3D input file, the second processor is the physical model itself (CODESA-3D) and the third one (*extract_solution*) reads a CODESA-3D output file containing the pressure heads of all the grid nodes and extract only nodal unknowns corresponding to the available measures allowing PEST to calculate residuals.

It should be stressed that the *ad hoc* tools developed to process data exchanged between models are strictly application dependent. Thus the choice of a specific parameter set strongly affects the needed translator programs.

```
echo ' run write_soil'
write_soil;
echo ' run codesa3d.r6k'
codesa3d.r6k;
echo ' run extract_solution'
extract_solution;
```

Figure 3. Batch file *doit* called by PEST as the embedded model consisting of 3 executables in succession.

In the following the specific problem under study and the data exchange between executables are described in the case of a coastal aquifer system where hydraulic permeability values are assumed as adjustable parameters and piezometric heads are the corresponding control measurements.

Synthetic observations as control data

To test the overall optimization procedure we used synthetic observations generated by the CODESA-3D model itself. The advantage of using synthetic observations instead of real measurements for the given problem is that we can test the optimization system without the system being clouded by issues related to failure of the groundwater model and available data to represent real-world problems. After the verification of the proposed methodology and implementation, the next step will be to actually calibrate a real-world problem.

The synthetic observations were obtained in the following way. We ran first a realistic case study (Oristano aquifer system, Italy [Lecca and Cau, 2004] and we recorded a subset of the corresponding model outputs (piezometric heads) in some selected points of the 3D aquifer grid. Hence we modified some initial model parameters (hydraulic permeability) editing the corresponding model input files. Then the optimization exercise was to verify model capability to find again the initial hydraulic permeability values (prior user modification) by means of calibrating the model against the recorded piezometric heads.

Oristano case study. We describe here only the characteristics of the case-study relevant to the test of the optimization model, all the other properties can be found in the cited bibliography. The 3D aquifer mesh, containing 20603 nodes and 108540 tetrahedra, was obtained by replicating vertically the land surface triangulation to form 10 layers of variable thickness. These layers discretize the three main hydrogeological units: upper (1) and lower (2) aquifers and the interbedded aquitard. The groundwater flow model was implemented using the original values (target values of parameters for the calibration test) of aquifer and aquitard hydraulic conductivity. These values are:

- upper aquifer horizontal (kh_aquifer1) and vertical (kv_aquifer1) hydraulic conductivity;
- lower aquifer horizontal (kh_aquifer2) and vertical (kv_aquifer2) hydraulic conductivity;
- aquitard isotropic hydraulic conductivity (k_aquitard).

After the first “synthetic run” (flow steady state) these values were modified to some initial values of the adjustable parameters as shown in Table 1.

hydraulic conductivity [m/s]	target value	initial value
kh_aquifer1	10^{-5}	10^{-4}
kv_aquifer1	10^{-6}	10^{-4}
k_aquitard	10^{-8}	10^{-7}
kh_aquifer2	10^{-5}	10^{-7}
kv_aquifer2	10^{-6}	10^{-5}

Table 1. Target and initial values of the adjustable parameters.

The CODESA-3D-PEST interface

To build up the CODESA-3D/PEST interface are required 3 types of input files:

- *Template files*, one for each model input file where adjustable parameters are located;
- *Instruction files*, one for each model output file where model-generated observations are located;
- *Control file*, a single file supplying PEST with information about names of template and instruction files, names of the corresponding model input and output files, problem size, control variables, initial parameter values, measurement values and weights.

Template files

For each input file containing adjustable parameters we must create a corresponding template file to instruct PEST on the names and the relative positions in the file of any of such adjustable parameters. This file is used by PEST to rewrite properly input files with the current best parameter set and re-run the embedded model. In our case study we ask PEST to adjust aquifer hydraulic conductivity values (k) written in the CODESA-3D input file 'soil' (Figure 4). The file contains many parameters as described in SWIMED deliverable D6 [Ababou and Al-Bitar, 2005] but the relevant ones are only 5 numbers corresponding to: kh_aquifer1, kv_aquifer1, kh_aquifer2, kv_aquifer2 and k_aquitard.

```

-10.          PMIN (m)
 3          IVGHU(0 VG, 1 XVG, 2 HU **n, 3 HU **G, 4 BC)
3.35 0.08 -3.0 VGN,VGRMC,VGPSAT
0.015 0. 0. -0.10 0.01 --- HUALFA,HUBETA,HUGAMA,HUPSIA,HUSWR
2.0          HUN
0. 0.          --- HUA,HUB
2.25 0.02 -0.25 BCBETA,BCRMC,BCPSAT
1.0E-05 1.0E-05 1.0E-06 1.0E-05 3.0E-01 --zone=1 nstr=1 phreatic aquifer
1.0E-05 1.0E-05 1.0E-06 1.0E-05 3.0E-01 -- 2 1
1.0E-05 1.0E-05 1.0E-06 1.0E-05 3.0E-01 -- 3 1

. . .
. . .
1.0E-08 1.0E-08 1.0E-08 1.0E-05 3.0E-01 -- 1 4 aquitard
. . .
. . .
1.0E-05 1.0E-05 1.0E-06 1.0E-05 3.0E-01 -- 1 10 semi-confined aquifer
1.0E-05 1.0E-05 1.0E-06 1.0E-05 3.0E-01 -- 2 10
1.0E-05 1.0E-05 1.0E-06 1.0E-05 3.0E-01 -- 3 10

kx ky kz S n

```

Figure 4. CODESA-3D input file *soil* containing the selected adjustable parameters.

To this end an intermediate input file (*Oristano-K.inp*) was created selecting from file *soil* only the relevant information to be read and re-written by PEST with the current optimal set. The file is shown in Figure 5. The corresponding template file (*Oristano-K.tpl*), which is shown in Figure 6, notifies to PEST that only 5 numbers separated by a special character ('#' is chosen here) are chosen as the calibration parameters. The intermediate input file is then pre-processed by the tool *write_soil* to rewrite the CODESA-3D input file *soil*.

```

10 3          nstrat, nzone
4          iaquitardo
1.0E-05 1.0E-06
1.0E-08
1.0E-05 1.0E-07

```

Figure 5. Example of intermediate input file (*Oristano-K.inp*) for which the template file of Figure 6 was build.

Preparation of a template file is a simple procedure. Comparing Figure 5 and 6 we see that it can be easily done using a text editor to replace chosen parameter values on a typical model input file by their respective parameter name and space identifiers. PEST is not interested at all in the other input parameters that are present in the intermediate file that are used by the pre-processor *write_soil* (Figure 3) to re-write the CODESA-3D input file soil. The '#' special character delimits the name of the variable and the range of its input format in the file.

```
ptf #          PEST Template File
10 3          nstrat, nzone
4            iaquitardo
#kh_aquifer1# #kv_aquifer1#
#k_aquitard #
#kh_aquifer2# #kv_aquifer2#
```

Figure 6. Example of PEST template file (*Oristano-K.tpl*) corresponding to the input file shown in Figure 5.

Instruction files

Of the possibly voluminous amount of information that a model may write to its output files, PEST is interested in only those outputs values for which corresponding field/laboratory data are available. For every model output file containing model-generated observations we must provide a template file in order to instruct PEST to read it.

```
flow: nodal pressure heads
this file is printed only if IPRT>=1
  0  0.0000E+00      NSTEP  TIME
0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00
. . . . .
. . . . .
1.119900E+02  1.064420E+02  1.008650E+02  7.925900E+01  8.166900E+01
7.791800E+01  7.984000E+01  1.101340E+02
  1  1.7000E+38      NSTEP  TIME
1.030000E+01  1.030000E+01  0.000000E+00  0.000000E+00  0.000000E+00
0.000000E+00  0.000000E+00 -1.119761E+00 -2.050025E-01  0.000000E+00
. . . . .
. . . . .
1.109743E+02  1.050539E+02  9.603888E+01  7.731216E+01  7.387076E+01
7.582153E+01  7.533511E+01  1.038622E+02
```

Figure 7. Example of CODESA-3D output file (*psi.out*) containing pressure head grid nodal values for each user-selected time step. Only few of the 41,225 file lines are shown here.

For a steady state flow simulation, we are interested in a single CODESA-3D output file (*psi.out*). This file (Figure 7) contains the pressure head of the 3D grid nodes (20603) for all time steps (in our case initial 0.0000E+00 and final 1.7000E+38 time).

```
soluzione al tempo 0.1699999976E+39 (steady state)
-1.42771E+01
-1.07762E-01
0.000000E+00
-3.40603E-01
. . .
. . .
1.42683E+02
1.36570E+02
1.08960E+02
```

Figure 8. Example of intermediate output file (*psi-control.out*) for which the instruction file of Figure 9 was build. In this specific example the file contains only 188 of the 41,225 lines of the corresponding CODESA-3d output file.


```

pif #
# solution #
11 (psi001) 1:15
11 (psi002) 1:15
11 (psi003) 1:15
11 (psi004) 1:15
. . .
. . .
11 (psi185) 1:15
11 (psi186) 1:15
11 (psi187) 1:15

```

Figure 9. Example of PEST instruction file (*Oristano-K.ins*) to read the intermediate output file shown in Figure 8.

Thus we first post-process this output, using the *extract_solution* tool (Figure 3), to write a small text file (Figure 8) containing only the simulated piezometric heads of the selected grid nodes (187) for which available field measurements exist. Then PEST is easily instructed to read this intermediate file by means of the PEST instruction file. The corresponding instruction file (Figure 9) contains a comment line (# solution #), the PEST instruction to move the file cursor of 1 line below at each time (line advance: 11), the name of the control variable (psi<>) and the maximum range of the output format which goes from column 1 to 15.

Control file

Once all the template and instruction file have been prepared a PEST control file must be assembled which brings it all together. An example of this file is given in Figure 10. The file is subdivided into sections as described below.

The “*control data*” section of the PEST control file is used to set internal array dimensions, tune and optimization options, and to set data output options. Variable **PESTMODE** is set to “estimation”. Other options are “regularization” and “prediction”. Variable **NPAR** is the total number of adjustable parameters, **NOBS** is the total number of available field observations, **NPRIOR** is the number of articles of prior information, **NTPLFILE** is the number of template files and **NINSFILE** is the number of instruction files. The meanings of the other control variables are explained in the PEST manual [Doherty, 2002].

Each adjustable parameter must belong to a “*parameter group*” since the input variables that define how derivatives are calculated pertain to the parameter group rather than individual parameter. The main characteristics set in this section are:

- the parameter group name (**PARGPNAME**);
- the increment type (**INCTYP**);
- the derivative increment (**DERINC**) used to perform forward-difference calculation;
- the derivative increment lower bound (**DERINCLB**);
- the type of finite difference scheme (**FORCEN**, forward/central);
- the maximum allowed multiplier for **DERINC** (**DERINCMUL**).

The “*parameter data*” section of the PEST control file is divided into two parts. The first part is related to the true adjustable parameters and the second is related to tied parameters⁵. If there are no tied parameters this last section is omitted. For instance, for the adjustable parameter **kh_aquifer1** we set a type log-transformed, factor-limited (greater than zero), with initial value **1.000000E-04** and lower and upper value **1.000000E-06** and **1.000000E-04**, respectively.

In the “*observation groups*” section of the PEST control file a name is supplied for every observation group. In our case we had only one set of observation named **obsgroup**.

For every observation cited in the PEST instruction file (Figure 9) there must be one line of data in the “*observation data*” section. The line contains 4 items: observation name, observation measurement value and (non-negative) weight, and observation group. If observations are of

⁵ Tied parameters are variables linked to adjustable parameters by means of a user-supplied mathematical equation.

different types, weights are vital in setting the relative importance of each measurement in the overall optimization process.

The “*model command line*” section of the PEST control file supplies the command (batch file **doit**, Figure 3) that PEST must use to run the embedded model.

```
pcf
* control data
restart  estimation
      5  187    5    0    1
      1    1 single point  1  0  0
JACFILE MESSFILE
      5.0  2.0  0.3  0.03    10
      3.0  3.0  0.001
      0.1
      30  0.01    3    3  0.01    3
RELPARSTP NRELPAR
      1    1    1
      ICOV ICOR IEIG
* parameter groups
kh_aquifer1 relative 0.01 0.0 switch 2.0 parabolic PARGPME INCTYP DERINC
DERINCBL FORCEN DERINCMUL NRELPAR
kv_aquifer1 relative 0.01 0.0 switch 2.0 parabolic
k_aquitard relative 0.01 0.0 switch 2.0 parabolic
kh_aquifer2 relative 0.01 0.0 switch 2.0 parabolic
kv_aquifer2 relative 0.01 0.0 switch 2.0 parabolic
* parameter data
kh_aquifer1 log factor 1.000000E-04 1.000000E-06 1.000000E-04
kh_aquifer1 1.0000 0.00000E+00 1
kv_aquifer1 log factor 1.000000E-04 1.000000E-06 1.000000E-04
kv_aquifer1 1.0000 0.00000E+00 1
k_aquitard log factor 1.000000E-07 1.000000E-10 1.000000E-07 k_aquitard
1.0000 0.00000E+00 1
kh_aquifer2 log factor 1.000000E-07 1.000000E-07 1.000000E-04
kh_aquifer2 1.0000 0.00000E+00 1
kv_aquifer2 log factor 1.000000E-05 1.000000E-08 1.000000E-05
kv_aquifer2 1.0000 0.00000E+00 1
* observation groups
obsgroup
* observation data
psi001 -1.42771E+01 1.0 obsgroup
psi002 -1.07747E-01 1.0 obsgroup
psi003 0.00000E+00 1.0 obsgroup
psi004 -3.40609E-01 1.0 obsgroup
psi005 -4.02249E+00 1.0 obsgroup
. . .
. . .
. . .
psi182 1.16930E+02 1.0 obsgroup
psi183 1.33566E+02 1.0 obsgroup
psi184 1.24396E+02 1.0 obsgroup
psi185 1.42683E+02 1.0 obsgroup
psi186 1.36570E+02 1.0 obsgroup
psi187 1.08960E+02 1.0 obsgroup
* model command line
doit
* model input/output
Oristano-K.tpl Oristano-K.inp
Oristano-K.ins psi-controllo.out
* prior information
```

Figure 10. Example of the PEST control file used for the Oristano case study.

The “*model input/output*” section contains two lines as specified by the control variables **NTPLFLE=1** and **NINSFLE=1**. The first line is the name of the PEST template file (Oristano-K.tpl) and the corresponding name of the model input file (Oristano-K.inp) to which the template is matched. The second line is the name of the PEST instruction file (Oristano-K.ins) and the corresponding name of the model output file (psi-controllo.out).

Since the variable **NPRIOR** in the “*control data*” section is equal to zero, PEST do not expect any article of prior information to be taken into account in the “*prior information*” section of the PEST control file.

Results

As PEST executes, it writes a detailed record of the parameter estimation process to the file <filenamebase>.rec (Figure 11). After echoing its input data, the program calculates the objective function arising from the initial parameter set. The starting value of the sum of squared weighted residuals (objective function) for our case-study was 12600. The optimization process is terminated after 9 optimization iterations since over the last 3 successive iterations the objective function values were within a relative distance prescribed by the control variable **PHIREDSTP = 0.01** as termination criteria. The total number of CODESA-3D calls to achieve convergence was 79.

```
PEST Version 5.50. Watermark Numerical Computing.

PEST is running in Parameter Estimation mode.

PEST run record: case Oristano-K
(See file Oristano-K.rec for full details.)

Model command line:
doit

Running model .....
esegue scrivi_soil
  legge dal file Oristano-K.inp
  scrive il file soil
esegue codesa3d.r6k
esegue scrivi_soluzione
  legge il file punti-misure.grid
  legge il file psi.out
  scrive il file psi-controllo.out
  Sum of squared weighted residuals (ie phi) =    12600.

OPTIMISATION ITERATION NO.      :      1
Model calls so far              :      1
Starting phi for this iteration:    12600.

Calculating Jacobian matrix: running model  5 times .....
. . .
  parameter "k_aquitard" frozen: - update vector out of bounds

  Lambda =    5.0000      ----->
  running model .....
    Phi =    6901.0      (    .548 of starting phi)

  Lambda =    2.5000      ----->
  running model .....
    Phi =    6901.7      (    .548 of starting phi)

  Lambda =    10.000      ----->
  running model .....
    Phi =    6900.3      (    .548 of starting phi)
```

(continued)

No more lambdas: relative phi reduction between lambdas less than .0300
Lowest phi this iteration: 6900.3
Maximum factor change: 3.000 ["kh_aquifer1"]
Maximum relative change: .9910 ["kv_aquifer2"]

OPTIMISATION ITERATION NO. : 2
Model calls so far : 9
Starting phi for this iteration: **6900.3**
All frozen parameters freed

Calculating Jacobian matrix: running model 5 times
Lambda = 10.000 ----->
running model
Phi = 887.89 (.129 of starting phi)

No more lambdas: phi is less than .3000 of starting phi
Lowest phi this iteration: 887.89
Maximum factor change: 3.000 ["kv_aquifer2"]
Maximum relative change: 2.000 ["kv_aquifer2"]

. . .
. . .

OPTIMISATION ITERATION NO. : 9
Model calls so far : 66
Starting phi for this iteration: 5.93705E-03

Calculating Jacobian matrix: running model 10 times
Lambda = 20.000 ----->
running model
Phi = 5.82846E-03 (.982 of starting phi)

Lambda = 10.000 ----->
running model
Phi = 5.85271E-03 (.986 of starting phi)

Lambda = 40.000 ----->
running model
Phi = 5.83299E-03 (.982 of starting phi)

No more lambdas: phi rising
Lowest phi this iteration: **5.82846E-03**
Maximum factor change: 1.002 ["kh_aquifer2"]
Maximum relative change: 2.2942E-03 ["kh_aquifer2"]

**Optimisation complete: relative parameter change less than 1.0000E-02
over 3 successive iterations.**

Total model calls: 79

Recording run statistics

See file **Oristano-K.rec** for full run details.
See file Oristano-K.sen for parameter sensitivities.
See file Oristano-K.seo for observation sensitivities.
See file Oristano-K.res for residuals.

Figure 11. Extract of the PEST run record file.

After completing the parameter estimation process, PEST prints the optimized parameter values with the 95% confidence limits (Figure 12). The estimated values of the hydraulic conductivities (red color in the same Figure) are practically identical to the target values of the synthetic run provided in Table 1 demonstrating the success of the optimization exercise.

After this information, the measured and the model-generated observation values are listed on the basis of the optimized parameter set along with the residuals. In addition a number of statistics pertaining to observation residuals are printed (same Figure). In our case-study the standard error of weighted residuals, calculated dividing the objective function value for the number of the degree of freedom of the system⁶, equals to 7.7461E-05.

OPTIMISATION RESULTS

Parameters ----->

Parameter	Estimated value	95% percent confidence limits	
		lower limit	upper limit
kh_aquifer1	9.999813E-06	9.999457E-06	1.000017E-05
kv_aquifer1	1.000315E-06	9.999998E-07	1.000631E-06
k_aquitard	9.999836E-09	9.999651E-09	1.000002E-08
kh_aquifer2	1.000003E-05	9.999973E-06	1.000009E-05
kv_aquifer2	9.999725E-07	9.999435E-07	1.000002E-06

. . .

Observations ----->

Observation	Measured value	Calculated value	Residual	Weight	Group
psi001	-14.2771	-14.2771	0.000000E+00	1.000	obsgroup
psi002	-.107747	-.107763	1.600000E-05	1.000	obsgroup
psi003	0.000000E+00	0.000000E+00	0.000000E+00	1.000	obsgroup
psi004	-.340609	-.340604	-5.000000E-06	1.000	obsgroup
. . .					
psi184	124.396	124.396	0.000000E+00	1.000	obsgroup
psi185	142.683	142.683	0.000000E+00	1.000	obsgroup
psi186	136.570	136.570	0.000000E+00	1.000	obsgroup
psi187	108.960	108.960	0.000000E+00	1.000	obsgroup

. . .

Analysis of residuals ----->

All residuals:-

Number of residuals with non-zero weight	=	187
Mean value of non-zero weighted residuals	=	-1.1150E-05
Maximum weighted residual [observation "psi037"]	=	3.0000E-05
Minimum weighted residual [observation "psi172"]	=	-1.0000E-03
Standard variance of weighted residuals	=	6.0002E-09
Standard error of weighted residuals	=	7.7461E-05

Figure 12. Some optimization results from the PEST run record file Oristano-K.rec.

Other important by-products of the parameter estimation process are the parameter covariance and the correlation coefficient matrices (Figure 13).

The *parameter covariance matrix* C is a symmetric square matrix with as many columns and rows as the number of adjustable parameters. The diagonal elements of the matrix are the variance (square of its standard deviation) of the adjustable parameters. The off diagonal elements are the covariance between parameter pairs x_i, x_j : $\sigma_{ij} = \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle$ where $\mu_i = \langle x_i \rangle$ and $\mu_j = \langle x_j \rangle$ are the means of x_i and x_j , respectively.

If we assume that the system under study can be described by the linear equation $Xb = c$ (*linearity assumption*), where X is a $m \times n$ matrix, b is a vector of order n which holds system parameters

⁶ Number of observations with non-zero weight.

and c is a vector of order m of model-generated observations, the covariance matrix is obtained by PEST as:

$$C(b) = \sigma^2 (X^T X)^{-1}$$

with $\sigma^2 = \Phi / (m - n)$ the reference variance and $\Phi = (c - Xb)^T (c - Xb)$ the objective function.

Notice that, even though the elements of c are assumed to be independent, C is not necessarily a diagonal matrix. In fact, in many problems parameters exhibit a strong correlation. In such cases some parameter variances (diagonal terms) may be large even though the objective function is reasonably low. In our case-study, since the adjustable parameters were log-transformed, the C coefficients pertain to the logarithm (to base 10) of the corresponding parameter. For instance (Figure 13) the variance of $\log(kh_aquifer1)$ equals to $.2439E-11$ hence the standard deviation is $1.56E-06$. The elements of the correlation coefficient matrix R are calculated as:

$$\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii} \sigma_{jj}}}$$

where the elements σ_{ij} represent the elements of the covariance matrix. The diagonal elements of the correlation matrix are always 1 while the off-diagonal elements range between -1 and 1. The closer are the off-diagonal elements to 1 or -1 the more highly are the respective parameters correlated. In our case-study the two highly correlated parameters were the aquitard and the phreatic aquifer vertical hydraulic conductivity values ($\rho=0.80$).

Parameter covariance matrix ----->					
	kh_aquifer1	kv_aquifer1	k_aquitard	kh_aquifer2	kv_aquifer2
kh_aquifer1	6.2439E-11	-1.5328E-10	4.1798E-12	-5.4260E-12	7.5010E-12
kv_aquifer1	-1.5328E-10	4.8884E-09	-2.2860E-10	1.4551E-11	-4.7520E-11
k_aquitard	4.1798E-12	-2.2860E-10	1.6808E-11	-9.2070E-13	-5.6515E-12
kh_aquifer2	-5.4260E-12	1.4551E-11	-9.2070E-13	1.6923E-12	-3.5831E-13
kv_aquifer2	7.5010E-12	-4.7520E-11	-5.6515E-12	-3.5831E-13	4.1514E-11
Parameter correlation coefficient matrix ----->					
	kh_aquifer1	kv_aquifer1	k_aquitard	kh_aquifer2	kv_aquifer2
kh_aquifer1	1.000	-.2774	.1290	-.5279	.1473
kv_aquifer1	-.2774	1.000	-.7975	.1600	-.1055
k_aquitard	.1290	-.7975	1.0000	-.1726	-.2140
kh_aquifer2	-.5279	.1600	-.1726	1.000	-4.2749E-02
kv_aquifer2	.1473	-.1055	-.2140	-4.2749E-02	1.0000

Figure 13. Parameter covariance and correlation coefficient matrices as additional information provided by PEST (Oristano-K.rec).

Most of the time consumed during each PEST iteration is devoted to calculation of the Jacobian matrix containing the derivatives of the observations with respect to adjustable parameters. During this process the model must be run at least $NPAR$ times⁷, where $NPAR$ is the number of adjustable parameters (5 in our case-study). From the Jacobian matrix some additional information can be obtained: the parameter and observation sensitivities.

Parameter sensitivity is a measure of composite changes in model outputs generated by a change in the value of the given parameter (Figure 14). Conversely observation sensitivity is a measure of all adjustable parameters change incurred by a change in the value of the given observation (Figure 15).

⁷ The number of model runs is equal to the number of adjustable parameters for forward finite difference scheme and twice the number of adjustable parameters for central finite difference scheme.

COMPLETION OF OPTIMISATION PROCESS				
Composite sensitivities for all observations/prior info ----->				
Number of observations with non-zero weight =		187		
Parameter name	Group	Current value	Sensitivity	Rel. Sensitivity
kh_aquifer1	kh_aquifer1	9.999813E-06	6.518657E-02	.325933
kv_aquifer1	kv_aquifer1	1.000315E-06	1.157415E-02	6.944330E-02
k_aquitard	k_aquitard	9.999836E-09	.197751	1.58201
kh_aquifer2	kh_aquifer2	1.000003E-05	.382093	1.91046
kv_aquifer2	kv_aquifer2	9.999725E-07	7.470985E-02	.448260

Figure 14. Part of the parameter sensitivity file (<filename>.sen). The last two columns (in red) are measures of the sensitivity of the given parameter to all model outputs.

Composite parameter sensitivities are useful for identifying those parameters that may degrade the performance of the optimization process through lack of sensitivity to model outputs. Observation sensitivities may be useful to identify observations particularly crucial to the inversion process because of their information content. This does not necessarily mean that the information is indispensable to the parameter estimation process because it can be also redundant as long as other observations at nearly the same time/space, with similar sensitivity, are included.

Observation	Group	Measured	Modelled	Sensitivity
psi001	obsgroup	-14.27710	-14.27710	3.537899
psi002	obsgroup	-.1077470	-.1077630	.5728126
psi003	obsgroup	0.0000000E+00	0.0000000E+00	0.0000000E+00
...				
psi185	obsgroup	142.6830	142.6830	.6736807
psi186	obsgroup	136.5700	136.5700	.4417134
psi187	obsgroup	108.9600	108.9600	.4582096

Figure 15. Part of the observation sensitivity file (<filename>.seo). The last column (in red) is a measure of the sensitivity of that measure to all adjustable parameters involved in the calibration process.

Concluding remarks and further applications

We have described here a composite model made of PEST free software tool and the hydrologic model CODESA-3D to perform optimal model parameter estimation. Our experience with PEST has been positive and successful for a steady-state groundwater problem using as control data synthetic observations generated by the model itself.

Further developments of the described procedure will be carried out for a real-world problem in a transient state simulation. Since CPU time will be much greater than in this case-study we plan to apply the PEST parallel option to speed up the calculations on networked PC.

Bibliography

1. Ababou, R. and A. Al-Bitar (editors), 2005. *Groundwater flow models* (SWIMED Deliverable 6, INCO Programme contract ICA3-CT2002-10004).
2. Doherty, J., *PEST Model-independent parameter estimation*, Watermark Numerical Computing, 2002.
3. Gambolati, G., M. Putti & C. Paniconi, 1999. *Three-dimensional model of coupled density-dependent flow and miscible transport in groundwater*. In Bear et al. (eds.) *Seawater intrusion in coastal aquifers: Concepts, Methods & Practices*, chap.10:315-362, Kluwer Academic, Dordrecht, The Netherlands.
4. Lecca, G., *Implementation and testing of the CODESA-3D model for density-dependent flow and transport problems in porous media*. CRS4-TECH-REP-00/40, Cagliari, Italy, 2000.
5. Lecca G. and P. Cau. *Stochastic Modeling for Seawater Intrusion Risk Assessment in Exploited Coastal Aquifers: the Oristano (Sardinia, Italy) Case Study*. In Cass T. Miller et al., editor, *Proceedings Of The XV International Conference On Computational Methods In Water Resources (CMWR XV)*, June 13-17, 2004 Chapel Hill, NC, USA., Jun 2004. Elsevier Science Ltd.
6. Lecca, G. and A. Larabi (editors), 2005. *Seawater intrusion models* (SWIMED Deliverable 7, INCO Programme contract ICA3-CT2002-10004).
7. Qahman et al., *Optimal and sustainable extraction of groundwater in coastal aquifers*. Stochastic Environmental Research and Risk Assessment, Springer Verlag 2005.